Multiple target tracking under occlusions using modified Joint Probabilistic Data Association

Xiufang Shi, Ye-Qiong Song, Zaiyue Yang, Jiming Chen

Multiple Target Tracking under Occlusions Using Modified Joint Probabilistic Data Association

Xiufang Shi*, Ye-Qiong Song‡, Zaiyue Yang* and Jiming Chen*
*State Key Lab. of Industrial Control Technology, Zhejiang University, China
‡LORIA, University of Lorraine, France
Email: xfshi.zju@gmail.com, Ye-Qiong.Song@loria.fr, yangzy@zju.edu.cn, jmchen@ieee.org

***Abstract*—The size of target will induce a degradation of tracking performance, which has been neglected for simplicity in most previous studies. In multiple target tracking, occlusions will be caused by target size effect, one target can become a moving obstacle blocking the direct channel between the anchor and another target. In this paper, the data association problem in multiple target tracking is investigated. To reduce the computational complexity of traditional Joint Probabilistic Data Association (JPDA) algorithm, a modified JPDA algorithm is proposed to execute data association in multiple target tracking by utilizing the information of occlusion conditions, which is identified by a three-step algorithm. Simulation results show that the proposed algorithm is with good tracking performance and low computational complexity.**

I. Introduction

Target tracking is essential in many applications, such as wildlife tracking, environment surveillance, submarine system, intelligent transportation [1]. In most existing studies of target tracking, the target is usually assumed to be a mass point, whose size is neglected to simplify the analysis. However, in many applications like vehicles tracking in battle field, target size cannot be neglected, as the neglect will induce a degradation of tracking performance. Therefore, some algorithms are proposed for target size estimation. In [2], the authors use networked binary sensors to estimate the shape of target and target size is estimated using ultrasonic sensors in [3].

In multiple target tracking, due to the impact of target size, during the movement of targets, one target may become an obstacle between an anchor and another target, blocking the direct channel between them, which is the so-called occlusion. In many tracking systems, e.g., radar tracking system, the sensor cannot get measurement originated from the occluded target. Many methods have been proposed to tackle occlusion problem [4]–[7]. In [4], [5], occlusion is solved by fusing multiple camera inputs. In [6], [7], appearance models are used to track occluded targets. Most of previously proposed methods are computer vision-based, which need specific instruments and are not suitable for other kinds of tracking systems, for instance, radar, ultrasonic-based tracking systems.

Different algorithms are designed for multiple target tracking under different conditions and purposes. For the purpose of multiple target localization, both the probability hypothesis density (PHD) filter [8] and compressive sensing-based algorithms [9], [10] can determine the number and positions of targets. However for the multiple target trajectories tracking, one of the most important problems is data association. A large amount of researchers have been working on this topic, and developed many algorithms to deal with data association problem, such as Joint Probabilistic Data Association (JPDA) [11], nearest neighbor (NN) [12], multiple hypothesis tracking (MHT) [13], Markov Chain Monte Carlo Data Association (MCMCDA) [14], etc. While in these algorithms, target size is neglected and occlusions are not considered.

In this paper, we will design a data association algorithm in multiple target tracking by taking advantage of the information of target size and occlusions to improve tracking performance. The contributions of our work are summarised as below:

- Multiple target tracking problem is investigated, in which, both the target size and occlusion conditions are taken into account.
- A modified JPDA (MJPDA) algorithm is proposed to improve the traditional JPDA by utilizing the occlusion information, which reduces the computational complexity.
- A three-step algorithm is developed to identify if one target is occluded by another, which is beneficial to reducing the number of joint events in data association algorithm.

The rest of this paper is organized as follows. Section II formulates the multiple target tracking problem. Traditional JPDA and our proposed MJPDA algorithms are introduced in Section III. Tracking performance of the proposed algorithm is verified by simulations in Section IV. Section V concludes this paper.

II. Problem Formulation

Assume there are T targets moving in a two dimensional field of interest (FOI), the state of target i at time t_k is $X_i(k) = [x_i(k),\ \dot{x}_i(k),\ y_i(k),\ \dot{y}_i(k),\ r_i]'$, where $[\quad]'$ denotes the transposition of a matrix or vector. $(x_i(k), y_i(k))$ is the coordinate of target i at time t_k, $\dot{x}_i(k)$ and $\dot{y}_i(k)$ are respectively the velocities of target i along x-axis and y-axis, r_i is the radius of target i. Here targets are approximated to be circles. In practice, most targets are of irregular shapes, however the target can be represented by a circle which covers the outline of the target. The motion model of target i is

$$X_i(k) = F_i X_i(k-1) + \omega_i(k), \quad i = 1, 2, \cdots, T \qquad (1)$$

where F_i is the state transition matrix, which can model different motions of targets, e.g., linear motion, circular motion, random walk, etc. $\omega_i(k)$ is the process noise, which is assumed to be zero-mean white Gaussian noise, with covariance matrix Q. As the target size is included in the state vector $X_i(k)$, it can be estimated simultaneously with target position.

There are N_a anchors deployed in the FOI, whose coordinates are known as (x_a, y_a), $a = 1, 2, \cdots, N_a$. At time t_k, anchor a can get distance measurement originated from target i which is represented as

$$z_{ai}(k) = d_{ai}(k) - r_i + v_a(k) \quad a = 1, 2, \cdots, N_a \tag{2}$$

where $d_{ai}(k) = \sqrt{(x_i(k) - x_a)^2 + (y_i(k) - y_a)^2}$ is the actual distance between anchor a and the center of target i. $v_a(k)$ is zero-mean white Gaussian noise.

In our considered multiple target tracking problem, each anchor will make an estimation individually, then the target state will be obtained by fusing the estimations on all these anchors. We will focus on the processing at a single anchor node a. To simplify the notation, we will drop the index a unless it is necessary to clarify. At time t_k, measurement vector at anchor a is $Z(k) = [z_1(k), \cdots, z_m(k), \cdots, z_M(k)]'$, where M is the number of measurements. Let P_D denote target detection probability and there exist Poisson distributed clutters with spatial density C, as a result, M might be unequal to target number T. Moreover, M is changeable from one time to the next. The origin of each measurements is uncertain. The key to this multiple target tracking problem is data association.

In our work, if the direct channel between anchor a and target i is occluded, there would be no measurement about target i at anchor a. Let s_{ai} denote the channel sight condition between anchor a and target i. If target i is occluded, $s_{ai} = 1$, otherwise $s_{ai} = 0$. As we take target size into consideration, during the movement of targets, one target can become a moving obstacle that blocks the direct channel between anchors and other targets, causing occlusions. For instance, in Fig. 1, the direct channel between anchor 3 and target 3 is occluded by target 2, then $s_{33} = 1$, anchor 3 cannot get distance measurement about target 3.

In most of the existing studies about data association, occlusion conditions are not considered. If the information of occlusion conditions is utilized, the traditional data association algorithms will be improved. In this paper, we will explore a method to take advantage of the information of target size and occlusion conditions to improve one of the most famous data association algorithms - JPDA.

III. Algorithm Design

In this section, we will first introduce the traditional JPDA, then we will take occlusions into consideration and propose a modified JPDA to improve the efficiency of traditional JPDA.

A. Traditional JPDA

In a JPDA filter, extended Kalman Filter (EKF) is utilized for target state prediction and estimation. Assume at time t_{k-1}, the estimated state and covariance of target i are $\hat{X}_i(k-1|k-1)$ and $P_i(k-1|k-1)$ respectively. Then at t_k, the predicted state $\hat{X}_i(k|k-1)$ and covariance $P_i(k|k-1)$ of target i are computed as

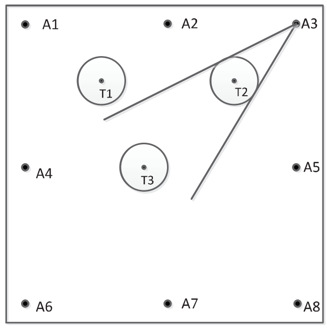

Fig. 1. T_1, T_2, T_3 stand for targets and $A_1 - A_8$ represent anchor nodes. Direct channel between anchor 3 and target 3 is occluded by target 2.

$$\hat{X}_i(k|k-1) = F_i \hat{X}_i(k-1|k-1) \tag{3}$$

$$P_i(k|k-1) = F_i P_i(k-1|k-1) F_i' + Q \tag{4}$$

Let

$$\hat{z}_i(k|k-1) = \sqrt{(x_i(k|k-1) - x_a)^2 + (y_i(k|k-1) - y_a)^2}$$

represent the predicted measurement from target i at anchor a. $\tilde{z}_{mi}(k) = z_m(k) - \hat{z}_i(k|k-1)$ denotes the residual error between measurement $z_m(k)$ and $\hat{z}_i(k|k-1)$. To simplify our notation, we will drop the time stamp k in the following description, i.e. $z_m(k)$ is simplified as z_m, etc. Based on predicted states obtained from (3), the nearly impossible measurements for each target are excluded by establishing a validation gate. The validation gate is an ellipsoid centered at the predicted measurement. If

$$\tilde{z}_{mi}' S_i^{-1} \tilde{z}_{mi} \leq g^2 \tag{5}$$

then z_m is in the validation gate of target i, indicating that z_m might come from target i, otherwise, z_m is excluded. In (5), g is a threshold deciding the size of validation gate. And $S_i = H_i P_i(k|k-1) H_i' + R$, where R is the covariance of measurements and H_i is the Jacobian matrix. Fig. 2 shows the validation gates of target 1 and target 2. From this figure, measurements z_1 and z_2 are in the validation gate of target 1, and measurements z_2 and z_3 are in the validation gate of target 2.

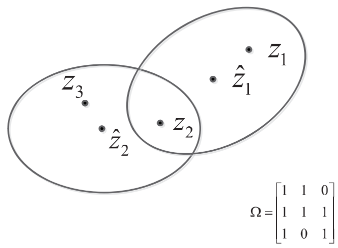

Fig. 2. Validation gates of target 1 and target 2.

The possible measurements for each target can be obtained via validation gates. Based on these results, a validation matrix is built up as follows

$$\Omega = \begin{bmatrix} w_{10} & w_{11} & \cdots & w_{1T} \\ w_{20} & w_{21} & \cdots & w_{2T} \\ \vdots & \vdots & \ddots & \vdots \\ w_{M0} & w_{M1} & \cdots & w_{MT} \end{bmatrix} \tag{6}$$

where

$$w_{mi} = \begin{cases} 1, & \tilde{z}'_{mi} S_i^{-1} \tilde{z}_{mi} \le g^2 \\ 0, & \tilde{z}'_{mi} S_i^{-1} \tilde{z}_{mi} > g^2 \end{cases}, \ i = 1, \cdots T \tag{7}$$

and if $i = 0$,

$$w_{m0} = \begin{cases} 1, & z_m \, comes\, from\, clutters \\ 0, & z_m \, comes\, from\, targets \end{cases} \tag{8}$$

Then the validation matrix Ω can be separated into several association matrices representing different joint events. The separation needs to follow two rules: 1) There is only one origin for each measurement; 2) One target can generate at most one measurement. In the example of Fig.2, Ω can be separated into 8 joint events $\{\chi_1, \chi_2, \cdots, \chi_8\}$, where

$$\chi_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \chi_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \chi_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \chi_4 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\chi_5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \chi_6 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \chi_7 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \chi_8 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Assume the number of joint events is N_e, from [11], the expression of probability of joint event χ_e is

$$P(\chi_e|Z) = \frac{C^{\varphi}}{c} \prod_{m:\tau_m=1} \frac{\exp[-\frac{1}{2}(\tilde{z}_{mi})' S_i^{-1} \tilde{z}_{mi}]}{(2\pi)^{M/2} |S_i|^{1/2}} \prod_{i:\delta_i=1} {P_D}^i \prod_{i:\delta_i=0} (1 - P_D^i), \quad e = 1, 2, \cdots, N_e \tag{9}$$

where φ and C are respectively the number and density of false measurements which are assumed to be Poisson distributed. c is a normalization constant. $\tilde{z}_{mi}$ represents the residual error that z_m is associated with target i in joint event χ_e. τ_m is a measurement association indicator in joint event χ_e, $\tau_m = 1$ indicates that measurement z_m is associated with a target, $\tau_m = 0$ indicates that measurement z_m is associate with clutters. δ_i is a target detection indicator in χ_e, $\delta_i = 1$ indicates that target i is detected, and $\delta_i = 0$ indicates that target i is not detected. P_D^i stands for the detection probability of target i.

Therefore, the probability that measurement z_m originated from target i, which is denoted as β_{mi}, is obtained by summing all joint events,

$$\beta_{mi} = \sum_{e=1}^{N_e} P(\chi_e|Z) w_{mi}, \quad \beta_{m0} = 1 - \sum_{i=1}^{T} \beta_{mi} \tag{10}$$

According to (10), association probabilities are used as joint event weights to combine target estimates under different association occasions. Estimated target states are computed by the method provided in [15].

B. Modified JPDA (MJPDA)

In the traditional JPDA, the computation complexity is $O(N_e(MT)^2)$, we can find that, joint events number N_e has a direct impact on the computational complexity. And N_e is decided by the validation matrix Ω. We will propose an algorithm called MJPDA to simplify Ω by using the information of target size and occlusion conditions between targets and anchors, which will reduce the computational complexity.

Specifically, N_e is decided by the number and location of 1-valued elements in Ω. If the number of 1-valued elements is reduced, thereby, the number of joint events will be reduced. The occlusion information will be utilized to reduce the number of elements valued with 1.

In Ω, the positions of 1 along mth row indicate the possible origins of z_m. We make a scan along each row of Ω, if there exist $w_{mi} = 1$ and $w_{mj} = 1$, z_m might come from target i or target j. As we consider the case that one target can be an obstacle between the anchor and another target, there would be no measurement about the occluded target. If target i or target j is occluded, we can directly set $w_{mi} = 0$ or $w_{mj} = 0$. Consequently, N_e will be reduced. Therefore, it is necessary to identify if there exists any target being occluded.

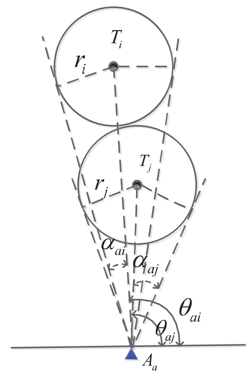

Fig. 3. Target i is occluded by target j, $s_{ai} = 1, s_{aj} = 0$

To identify sight conditions between targets and anchors, a geometry-based algorithm is developed. As shown in Fig. 3,

$$\theta_{ai} = \arctan(\frac{y_i - y_a}{x_i - x_a}), \alpha_{ai} = \arcsin(\frac{r_i}{d_{ai}})$$
$$\theta_{aj} = \arctan(\frac{y_j - y_a}{x_j - x_a}), \alpha_{aj} = \arcsin(\frac{r_j}{d_{aj}})$$

Let $\phi_{ai}^{\min} = \theta_{ai} - \alpha_{ai}$, $\phi_{ai}^{\max} = \theta_{ai} + \alpha_{ai}$, $\phi_{aj}^{\min} = \theta_{aj} - \alpha_{aj}$ and $\phi_{aj}^{\max} = \theta_{aj} + \alpha_{aj}$, then the range of bearing angles of target i and target j relative to anchor a are respectively $(\phi_{ai}^{\min}, \phi_{ai}^{\max})$ and $(\phi_{aj}^{\min}, \phi_{aj}^{\max})$. If $d_{ai} > d_{aj}$, $\phi_{ai}^{\min} \ge \phi_{aj}^{\min}$ and $\phi_{ai}^{\max} \le \phi_{aj}^{\max}$, then target i is occluded by target j. As the actual positions of targets are unknown, we should identify sight conditions from the point of statistical probability. Let P_{ija} denote the probability that target i is occluded by target j, it should be computed as

$$P_{ija} = P(d_{ai} > d_{aj}, \phi_{ai}^{\min} \ge \phi_{aj}^{\min}, \phi_{ai}^{\max} \le \phi_{aj}^{\max}) \tag{11}$$

Considering that all of these three conditions in (11) are functions of actual states of both targets, they are not independent of each other statistically. It is of high complexity to compute probability in (11). A three-step algorithm is designed to identify if target i is occluded by target j. Three conditions in (11) will be checked one by one, if one of them is not satisfied, the probability computation will not be continued, which reduces the computational complexity, compared with the joint probability computation with three conditions.

Algorithm 1 : Three-step algorithm

1: Compute $P(d_{ai} > d_{aj})$, if $P(d_{ai} > d_{aj}) \geq P_d$, goto step 2, else, exit.
2: Compute $P(\phi_{ai}^{\min} \geq \phi_{aj}^{\min})$, if $P(\phi_{ai}^{\min} \geq \phi_{aj}^{\min}) \geq P_{\phi}^{\min}$, goto step 3, else, exit.
3: Compute $P(\phi_{ai}^{\max} \leq \phi_{aj}^{\max})$, if $P(\phi_{ai}^{\max} \leq \phi_{aj}^{\max}) \geq P_{\phi}^{\max}$, set $w_{mi} = 0$, else, exit.

In Algorithm 1, P_d, $P_{\phi}^{\min}$ and $P_{\phi}^{\max}$ are thresholds for probabilities under three corresponding conditions. $P(d_{ai} > d_{aj})$, $P(\phi_{ai}^{\min} \geq \phi_{aj}^{\min})$ and $P(\phi_{ai}^{\max} \leq \phi_{aj}^{\max})$ are computed as follows.

1) $P(d_{ai} > d_{aj})$: As d_{ai} is nonlinear with $X_i(k)$, a linearization is made to d_{ai} as a normal EKF usually does,

$$d_{ai} \approx \hat{d}_{ai} + H_{d_{ai}}(X_i(k) - \hat{X}_i(k|k-1))$$

where $\hat{d}_{ai} = \hat{z}_i(k|k-1)$, $H_{d_{ai}}$ is Jacobian matrix computed as

$$H_{d_{ai}} = \begin{bmatrix} \frac{\partial d_{ai}}{\partial x_i(k)} & \frac{\partial d_{ai}}{\partial \dot{x}_i(k)} & \frac{\partial d_{ai}}{\partial y_i(k)} & \frac{\partial d_{ai}}{\partial \dot{y}_i(k)} & \frac{\partial d_{ai}}{\partial r_i} \end{bmatrix}_{X_i(k)=\hat{X}_i(k|k-1)} \tag{12}$$

where

$$\frac{\partial d_{ai}}{\partial x_i(k)} = \frac{x_i(k) - x_a}{\sqrt{(x_i(k) - x_a)^2 + (y_i(k) - y_a)^2}}$$

$$\frac{\partial d_{ai}}{\partial y_i(k)} = \frac{y_i(k) - y_a}{\sqrt{(x_i(k) - x_a)^2 + (y_i(k) - y_a)^2}}$$

$$\frac{\partial d_{ai}}{\partial \dot{x}_i(k)} = 0, \quad \frac{\partial d_{ai}}{\partial \dot{y}_i(k)} = 0, \quad \frac{\partial d_{ai}}{\partial r_i} = 0$$

State of target i satisfies $X_i(k) \sim N(\hat{X}_i(k|k-1), P_i(k|k-1))$, then d_{ai} follows

$$d_{ai} \sim N(\hat{d}_{ai}, H_{d_{ai}} P_i(k|k-1) H'_{d_{ai}}) \tag{13}$$

Similarly, d_{aj} follows

$$d_{aj} \sim N(\hat{d}_{aj}, H_{d_{aj}} P_j(k|k-1) H'_{d_{aj}}) \tag{14}$$

Because states of target i and target j are independent from each other, we can conclude (15). $d_{ai} - d_{aj}$ follows a normal distribution, then $P(d_{ai} > d_{aj})$ can be easily obtained by computing the cumulative distribution functions (CDF) of a normal distribution, namely,

$$P(d_{ai} > d_{aj}) = P(d_{ai} - d_{aj} > 0) \tag{16}$$

2) $P(\phi_{ai}^{\min} \geq \phi_{aj}^{\min})$: Similar with the computation of $P(d_{ai} > d_{aj})$, $P(\phi_{ai}^{\min} \geq \phi_{aj}^{\min})$ is computed as

$$P(\phi_{ai}^{\min} \geq \phi_{ja}^{\min}) = P(\phi_{ai}^{\min} - \phi_{aj}^{\min} \geq 0) \tag{17}$$

After linearization, $\phi_{ai}^{\min}$ is approximated as

$$\phi_{ai}^{\min} \approx \hat{\phi}_{ai}^{\min} + H_{\phi_{ai}}^{\min}(X_i(k) - \hat{X}_i(k|k-1))$$

where $\hat{\phi}_{ai}^{\min} = \hat{\theta}_{ai} - \hat{\alpha}_{ai}$ and $H_{\phi_{ai}}^{min}$ is expressed as

$$H_{\phi_{ai}}^{\min} = \begin{bmatrix} \frac{\partial \phi_{ai}^{\min}}{\partial x_i(k)} & \frac{\partial \phi_{ai}^{\min}}{\partial \dot{x}_i(k)} & \frac{\partial \phi_{ai}^{\min}}{\partial y_i(k)} & \frac{\partial \phi_{ai}^{\min}}{\partial \dot{y}_i(k)} & \frac{\partial \phi_{ai}^{\min}}{\partial r_i} \end{bmatrix}_{X_i(k)=\hat{X}_i(k|k-1)} \tag{18}$$

where

$$\frac{\partial \phi_{ai}^{\min}}{\partial x_i(k)} = \frac{-(y_i(k) - y_a)}{d_{ai}^2} - \frac{-2r_i(x_i(k) - x_a)}{d_{ai}\sqrt{d_{ai}^2 - r_i^2}}$$

$$\frac{\partial \phi_{ai}^{\min}}{\partial y_i(k)} = \frac{x_i(k) - x_a}{d_{ai}^2} - \frac{-2r_i(y_i(k) - y_a)}{d_{ai}\sqrt{d_{ai}^2 - r_i^2}}$$

$$\frac{\partial \phi_{ai}^{\min}}{\partial \dot{x}_i(k)} = 0, \quad \frac{\partial \phi_{ai}^{\min}}{\partial \dot{y}_i(k)} = 0, \quad \frac{\partial \phi_{ai}^{\min}}{\partial r_i} = \frac{1}{\sqrt{d_{ai}^2 - r_i^2}}$$

Then

$$\phi_{ai}^{\min} \sim N(\hat{\phi}_{ai}^{\min}, H_{\phi_{ai}}^{\min} P_i(k|k-1) H_{\phi_{ai}}^{\min'}) \tag{19}$$

Similarly,

$$\phi_{aj}^{\min} \sim N(\hat{\phi}_{aj}^{\min}, H_{\phi_{aj}}^{\min} P_j(k|k-1) H_{\phi_{aj}}^{\min'}) \tag{20}$$

As target i and target j are independent from each other, therefore (21) is concluded. Consequently, $P(\phi_{ai}^{\min} \geq \phi_{aj}^{\min})$ is obtained.

3) $P(\phi_{ai}^{\max} \leq \phi_{aj}^{\max})$: $P(\phi_{ai}^{\max} \leq \phi_{aj}^{\max})$ can be computed in the same way with $P(d_{ai} > d_{aj})$ and $P(\phi_{ai}^{\min} \geq \phi_{aj}^{\min})$.

The overall MJPDA algorithm is depicted in Algorithm 2.

Algorithm 2 :MJPDA

```
Build up validation matrix Ω as JPDA in (6).
for m = 1 : M do
  for i = 1 : T do
    for j = i : T do
      if w_mi = 1 and w_mj = 1 then
        Reset w_mi and w_mj using Algorithm 1.
      end if
    end for
  end for
end for
Separate modified Ω into N_e joint events.
Execute association probabilities computation and state estimation as JPDA.
```

IV. Simulation Results

Extensive simulations are conducted to evaluate the performance of the proposed algorithm. First, the influence of target size on the tracking performance is analyzed. Then, we will compare the performance of MJPDA and JPDA. In the end, the effect of target number on the computational complexity is illustrated. Moreover, to illustrate the robustness of the proposed algorithm to target motion , in the following simulation parts, we make targets to do different kinds of movements, i.e., linear movement, circular movement and random walk.

A. Influence of Target Size

In this paper we take target size into consideration, which is included in the vector of target state as r_i. The estimation process is the same with [3], which is applied to both JPDA and MJPDA. To illustrate the influence of target size, we just compare the tracking performance with and without target size considered both in JPDA.

There are 8 anchors located in a 200×200 FOI, with coordinates as (-100,-100), (0,-100), (100,-100), (-100, 0), (100, 0), (-100,100), (0,100), (100,100). There are two targets moving in the FOI. The size of these two targets $r = 5$. The clutter density is 1×10^{-5}. In the tracking algorithm, validation gate indicator $g = 5$, target detection probability $P_D = 0.99$. Fig.4 shows the actual trajectories of these two targets and Fig.5 shows estimated position errors with two different algorithms, one is JPDA with target size neglected, the other is JPDA with target size considered. Obviously, if target size is neglected, tracking performance is much worse than that with target size considered. It is of great necessity to take target size into account.

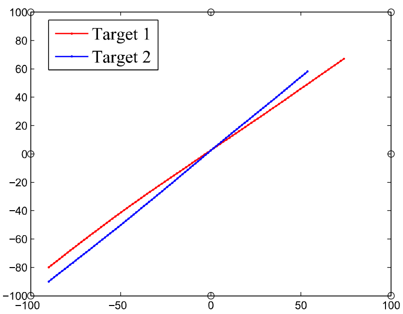

Fig. 4. Actual trajectories of two targets.

B. Tracking Performance Comparison

Let four targets make a circular movement as shown in Fig.6. The deployment of anchors and parameter settings

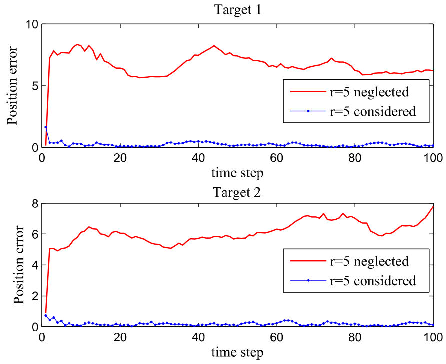

Fig. 5. Estimated position errors with target size neglected and considered.

are the same with section IV-A. We adopt two algorithms, i.e., MJPDA and JPDA to track these four targets. Fig. 7 compares root mean square errors (RMSEs) of four targets' position estimation when using MJPDA and JPDA. RMSE is computed as $RMSE_i(k) = \sqrt{\frac{1}{N_r}\sum_{n=1}^{N_r} ||\hat{p}_i(k) - p_i(k)||^2}$, where $\hat{p}_i(k) = [\hat{x}_i(k|k), \hat{y}_i(k|k)]$ and $p_i(k) = [x_i(k), y_i(k)]$. N_r is the number of Monte Carlo runs of tracking one trajectory. Here we set $N_r = 100$. We can see that, tracking accuracy of these two algorithms are almost the same. Fig.8 illustrates the average number of joint events by MJPDA and JPDA when executing a single run with 100 time steps. N_e by MJPDA is obviously lower than JPDA, namely, computational complexity of MJPDA is much lower than JPDA.

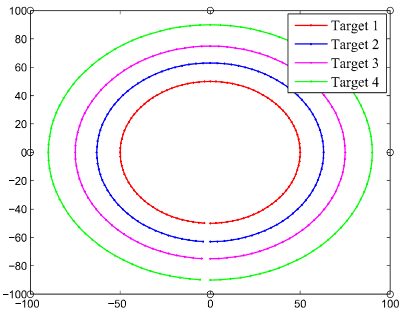

Fig. 6. Actual trajectories of four targets.

C. Influence of Target Number

To analyze the influence of target number on the performance of tracking algorithms, different numbers of targets are tracked by MJPDA and JPDA respectively. Table I compares the tracking performance of MJPDA and JPDA under 4,5,6,7

$$d_{ai} - d_{aj} \sim N(\hat{d}_{ai} - \hat{d}_{aj}, H_{d_{ai}} P_i(k|k-1) H'_{d_{ai}} + H_{d_{aj}} P_j(k|k-1) H'_{d_{aj}}) \tag{15}$$

$$\phi_{ai}^{\min} - \phi_{aj}^{\min} \sim N(\hat{\phi}_{ai}^{\min} - \hat{\phi}_{aj}^{\min}, H_{\phi_{ai}}^{\min} P_i(k|k-1) H_{\phi_{ai}}^{\min\prime} + H_{\phi_{aj}}^{\min} P_j(k|k-1) H_{\phi_{aj}}^{\min\prime}) \tag{21}$$

TABLE I
PERFORMANCE COMPARISON UNDER DIFFERENT TARGET NUMBERS

Target number	Averaged N_e by MJPDA	Averaged N_e by JPDA	Averaged RMSE by MJPDA	Averaged RMSE by JPDA
4	140.3469	154.9714	0.3532	0.3429
5	285.4069	338.5122	0.3533	0.3350
6	561.0216	709.9033	0.3860	0.3512
7	1.1671e+003	1.5954e+003	0.4337	0.3555

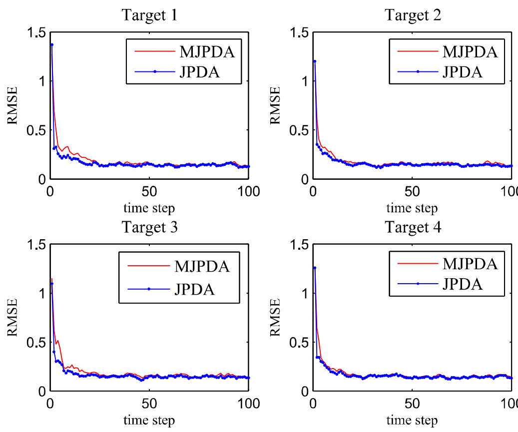

Fig. 7. RMSEs of four targets when using MJPDA and JPDA.

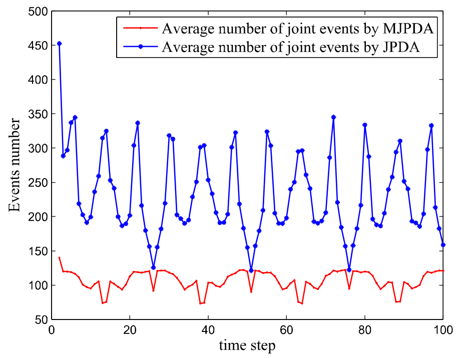

Fig. 8. Average number of joint events by MJPDA and JPDA.

targets, which make random walks. From this table, we can see that, with the increase of target number, the number of joint events will grow in a sharp speed, which will become the primary factor affecting the tracking performance. Even though tracking accuracy of JPDA is slightly higher than MJPDA, joint events number of MJPDA is much lower than that of JPDA. The computation complexity of MJPDA is much lower than JPDA. It can be concluded that tracking performance of MJPDA is better than JPDA.

V. CONCLUSION

In this paper, a multiple target tracking problem is investigated. By taking target size into consideration, the tracking accuracy is improved. Due to the target size effect, there will be occlusions during targets' movement. An algorithm is designed to identify the occlusion conditions. Utilizing the information of occlusion conditions in data association, the computational complexity of traditional JPDA algorithm can be reduced by the proposed MJPDA algorithm. Extensive simulations are conducted to verify the performance of our proposed algorithm, which show that MJPDA is with good tracking performance and low computational complexity.

ACKNOWLEDGMENT

This work is partly supported by 973 Program under Grant 2013CB329503, NSFC under Grants 61061130563 and 61371159. This work is also supported by the French National Research Agency (ANR 2010 INTB 0206 01).

REFERENCES

[1] Y. Bar-Shalom, "Multitarget-multisensor tracking: Applications and advances. volume iii," *Norwood, MA, Artech House, Inc.*, 2000.

[2] H. Saito, S. Shimogawa, S. Shioda, and J. Harada, "Shape estimation using networked binary sensors," in *Proceedings of IEEE INFOCOM*, 2009, pp. 2901–2905.

[3] J. Wang, P. Cheng, J. Chen, Y. Sun, and X. Shen, "Target tracking with size estimation in wireless sensor networks," in *Proceedings of IEEE WCSP*, 2011, pp. 1–5.

[4] S. L. Dockstader and A. M. Tekalp, "Multiple camera tracking of interacting and occluded human motion," *Proceedings of the IEEE*, vol. 89, no. 10, pp. 1441–1455, 2001.

[5] T.-H. Chang, S. Gong, and E.-J. Ong, "Tracking multiple people under occlusion using multiple cameras." in *BMVC*, 2000, pp. 1–10.

[6] Y. Wu, T. Yu, and G. Hua, "Tracking appearances with occlusions," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2003, pp. I–789.

[7] A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti, and R. Bolle, "Appearance models for occlusion handling," *Image and Vision Computing*, vol. 24, no. 11, pp. 1233–1243, 2006.

[8] Z. Shi, Y. Liu, S. Hong, J. Chen, and X. Shen, "POSE: Design of hardware-friendly particle-based observation selection PHD filter," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 4, pp. 1944–1956, 2014.

[9] K. Xin, P. Cheng, and J. Chen, "Multi-target localization in wireless sensor networks: a compressive sampling-based approach," *Wireless Communications and Mobile Computing*, 2013.

[10] J. Wang, D. Fang, X. Chen, Z. Yang, T. Xing, and L. Cai, "LCS: Compressive sensing based device-free localization for multiple targets in sensor networks," in *IEEE INFOCOM*, 2013, pp. 145–149.

[11] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE Journal of Oceanic Engineering*, vol. 8, no. 3, pp. 173–184, 1983.

[12] X. Rong Li and Y. Bar-Shalom, "Tracking in clutter with nearest neighbor filters: analysis and performance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 3, pp. 995–1010, 1996.

[13] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

[14] L. Chen, M. Cetin, and A. S. Willsky, "Distributed data association for multi-target tracking in sensor networks," 2005.

[15] Y. Bar-Shalom and E. Tse, "Tracking in a cluttered environment with probabilistic data association," *Automatica*, vol. 11, no. 5, pp. 451–460, 1975.